

**WHAT IS CLAIMED IS:**

- 1 1. A computer implemented method for asymmetrically  
2 multithreading tasks in a computer environment that  
3 includes a plurality of dissimilar processors, said  
4 method comprising:  
5 scheduling operation of a plurality of first tasks  
6 adapted to be executed by a first type of processor  
7 from the plurality of dissimilar processors, the  
8 scheduling performed by a first scheduler that  
9 maintains a first run queue that includes data  
10 corresponding to the first tasks; and  
11 scheduling operation of a plurality of second tasks  
12 adapted to be executed by a second type of processor  
13 from the plurality of dissimilar processors, the  
14 scheduling performed by a second scheduler that  
15 maintains a second run queue that includes data  
16 corresponding to the second tasks.
- 1 2. The method as described in claim 1 wherein the first  
2 scheduler and the second scheduler are located on the  
3 first type of processor.
- 1 3. The method as described in claim 2 wherein the  
2 scheduling operation of the plurality of first tasks  
3 is asymmetric to the scheduling operation of the  
4 plurality of second tasks.
- 1 4. The method as described in claim 2 further comprising:  
2 wherein a first identifier space corresponds to the  
3 first type of processor and wherein a second  
4 identifier space corresponds to the second type of  
5 processor; and

6 wherein a first task list corresponds to the first  
7 type of processor and wherein a second task list  
8 corresponds to the second type of processor.

1 5. The method as described in claim 2 wherein the  
2 scheduling operation of the plurality second tasks  
3 further comprises:  
4 receiving a new task from the plurality of second  
5 tasks;  
6 identifying new task attributes corresponding to the  
7 new task;  
8 comparing the new task attributes with one or more  
9 scheduled task attributes, the scheduled task  
10 attributes corresponding to one or more scheduled  
11 tasks that are included in the second run queue; and  
12 performing the scheduling of the new task based upon  
13 the comparing.

1 6. The method as described in claim 5 wherein at least  
2 one of the new task attributes are selected from the  
3 group consisting of a policy and a priority.

1 7. The method as described in claim 2 further comprising:  
2 informing the second type of processor to load one of  
3 the second tasks in response to the scheduling.

1 8. The method as described in claim 1 wherein the  
2 computer environment includes a plurality of second  
3 type of processors, and wherein the second scheduler  
4 maintains a plurality of second run queues, each of

5 the plurality of second run queues corresponding to  
6 each of the plurality of second type of processors.

1 9. The method as described in claim 8 wherein the  
2 scheduling operation of the plurality of second tasks  
3 further comprises:  
4 receiving a new task from the plurality of second  
5 tasks  
6 analyzing a plurality of workloads that correspond to  
7 the plurality of second type of processors;  
8 identifying an available second type of processor from  
9 the plurality of second type of processors in response  
10 to the analyzing; and  
11 including the new task in the second run queue from  
12 the plurality of second run queues that corresponds to  
13 the available second type of processor.

1 10. The method as described in claim 1 wherein the first  
2 type of processor is a processing unit and wherein the  
3 second type of processor is a synergistic processing  
4 unit.

5 11. An information handling system comprising:  
6 a plurality of dissimilar processors;  
7 a memory accessible by the plurality of dissimilar  
8 processors;  
9 one or more nonvolatile storage devices accessible by  
10 the plurality of dissimilar processors; and

11 an asymmetric multithreading tasking tool for  
12 scheduling tasks, the asymmetric multithreading  
13 tasking tool comprising software code effective to:

14 schedule operation of a plurality of first  
15 tasks adapted to be executed by a first type  
16 of processor from the plurality of  
17 dissimilar processors, the scheduling  
18 performed by a first scheduler that  
19 maintains a first run queue located on the  
20 memory that includes data corresponding to  
21 the first tasks; and

22 schedule operation of a plurality of second  
23 tasks adapted to be executed by a second  
24 type of processor from the plurality of  
25 dissimilar processors, the scheduling  
26 performed by a second scheduler that  
27 maintains a second run queue located on the  
28 memory that includes data corresponding to  
29 the second tasks.

1 12. The information handling system as described in claim  
2 10 wherein the first scheduler and the second  
3 scheduler are located on the first type of processor.

1 13. The information handling system as described in claim  
2 12 wherein the scheduling operation of the plurality  
3 of first tasks is asymmetric to the scheduling  
4 operation of the plurality of second tasks.

1 14. The information handling system as described in claim  
2 12 wherein the software code is further effective to:

3 wherein a first identifier space corresponds to the  
4 first type of processor and wherein a second  
5 identifier space corresponds to the second type of  
6 processor; and

7 wherein a first task list corresponds to the first  
8 type of processor and wherein a second task list  
9 corresponds to the second type of processor.

1 15. The information handling system as described in claim  
2 12 wherein the software code is further effective to:  
3 receive a new task from the plurality of second tasks;  
4 identify new task attributes corresponding to the new  
5 task;  
6 compare the new task attributes with one or more  
7 scheduled task attributes, the scheduled task  
8 attributes corresponding to one or more scheduled  
9 tasks that are included in the second run queue; and  
10 perform the scheduling of the new task based upon the  
11 comparing.

1 16. The information handling system as described in claim  
2 15 wherein at least one of the new task attributes are  
3 selected from the group consisting of a policy and a  
4 priority.

1 17. The information handling system as described in claim  
2 12 wherein the software code is further effective to:  
3 inform the second type of processor to load one of the  
4 second tasks in response to the scheduling.

1 18. The information handling system as described in claim  
2 10 wherein the computer environment includes a  
3 plurality of second type of processors, and wherein  
4 the second scheduler maintains a plurality of second  
5 run queues, each of the plurality of second run queues  
6 corresponding to each of the plurality of second type  
7 of processors.

1 19. The information handling system as described in claim  
2 18 wherein the software code is further effective to:  
3 receive a new task from the plurality of second tasks  
4 analyze a plurality of workloads that correspond to  
5 the plurality of second type of processors;  
6 identify an available second type of processor from  
7 the plurality of second type of processors in response  
8 to the analyzing; and  
9 include the new task in the second run queue from the  
10 plurality of second run queues that corresponds to the  
11 available second type of processor.

1 20. The information handling system as described in claim  
2 10 wherein the first type of processor is a processing  
3 unit and wherein the second type of processor is a  
4 synergistic processing unit.

5 21. A computer program product stored on a computer  
6 operable media for asymmetrically multithreading tasks  
7 in a computer environment that includes a plurality of  
8 dissimilar processors, said computer program product  
9 comprising:

10 means for scheduling operation of a plurality of first  
11 tasks adapted to be executed by a first type of  
12 processor from the plurality of dissimilar processors,  
13 the scheduling performed by a first scheduler that  
14 maintains a first run queue that includes data  
15 corresponding to the first tasks; and

16 means for scheduling operation of a plurality of  
17 second tasks adapted to be executed by a second type  
18 of processor from the plurality of dissimilar  
19 processors, the scheduling performed by a second  
20 scheduler that maintains a second run queue that  
21 includes data corresponding to the second tasks.

1 22. The computer program product as described in claim 21  
2 wherein the first scheduler and the second scheduler  
3 are located on the first type of processor.

1 23. The computer program product as described in claim 22  
2 wherein the scheduling operation of the plurality of  
3 first tasks is asymmetric to the scheduling operation  
4 of the plurality of second tasks.

1 24. The computer program product as described in claim 22  
2 further comprising:  
3 wherein a first identifier space corresponds to the  
4 first type of processor and wherein a second  
5 identifier space corresponds to the second type of  
6 processor; and

7 wherein a first task list corresponds to the first  
8 type of processor and wherein a second task list  
9 corresponds to the second type of processor.

1 25. The computer program product as described in claim 22  
2 wherein the scheduling operation of the plurality  
3 second tasks further comprises:  
4 means for receiving a new task from the plurality of  
5 second tasks;  
6 means for identifying new task attributes  
7 corresponding to the new task;  
8 means for comparing the new task attributes with one  
9 or more scheduled task attributes, the scheduled task  
10 attributes corresponding to one or more scheduled  
11 tasks that are included in the second run queue; and  
12 means for performing the scheduling of the new task  
13 based upon the comparing.

1 26. The computer program product as described in claim 25  
2 wherein at least one of the new task attributes are  
3 selected from the group consisting of a policy and a  
4 priority.

1 27. The computer program product as described in claim 22  
2 further comprising:  
3 means for informing the second type of processor to  
4 load one of the second tasks in response to the  
5 scheduling.

1 28. The computer program product as described in claim 21  
2 wherein the computer environment includes a plurality  
3 of second type of processors, and wherein the second  
4 scheduler maintains a plurality of second run queues,  
5 each of the plurality of second run queues



6 corresponding to each of the plurality of second type  
7 of processors.

1 29. The computer program product as described in claim 28  
2 wherein the scheduling operation of the plurality of  
3 second tasks further comprises:  
4 means for receiving a new task from the plurality of  
5 second tasks

6 means for analyzing a plurality of workloads that  
7 correspond to the plurality of second type of  
8 processors;

9 means for identifying an available second type of  
10 processor from the plurality of second type of  
11 processors in response to the analyzing; and

12 means for including the new task in the second run  
13 queue from the plurality of second run queues that  
14 corresponds to the available second type of processor.

1 30. The computer program product as described in claim 21  
2 wherein the first type of processor is a processing  
3 unit and wherein the second type of processor is a  
4 synergistic processing unit.